

**TeTe - Breast Cancer Prediction model**



Naam: Amaryllis Lee  
Datum: 29 juni 2020

# Inhoudsopgave

[**Probleembeschrijving**](#_3dy6vkm) **2**

[**Eisen**](#_1t3h5sf) **3**

[**Algoritme**](#_4d34og8) **4**

# **Probleembeschrijving**

Borstkanker heeft in de laatste jaren wereldwijd een grote impact veroorzaakt op zowel traumatisch - als financieel niveau. In Nederland krijgt 1 op de 7 vrouwen borstkanker. In 2019 kregen er 14.940 mensen borstkanker terwijl in 2018 3.088 mensen hiervan stierven. En er wordt steeds gezocht via onderzoeken en AI-technieken hoe ze de diagnose nauwkeurig of exact kan doen zodat de behandeling en de gevolgen hiervan minder ingrijpend.

Het doel van dit algoritme is om een voorspellingsmodel te ontwikkelen voor de diagnose van borstkanker voor medische klinieken. Waarom is dit van belang? In de praktijk worden vrouwen vaak verkeerd gediagnosticeerd. De diagnose kan zijn kwaadaardig terwijl in realiteit het goedaardig is of de diagnose is goedaardig terwijl het kwaadaardig is in realiteit. Beide gevallen hebben grote gevolgen voor de patiënt. Indien de borstkanker kwaadaardig is en in realiteit het goedaardig is, is het gevolg dat de patiënt een lange en ingrijpend behandeling die pijnlijk en tramautisch is, moet volgen. Andersom indien de borstkanker kwaadaardig is en de diagnose zegt goedaardig, zal de patiënt niet een behandeling hiervoor krijgen.

Daarom is het van belang dat de nauwkeurigheid van de diagnose bij borstkanker (bijna) exact is. Voor deze reden wordt gewerkt aan een voorspellingsmodel (applicatie) met een grote nauwkeurigheid voor borstkanker. In de literatuur komen verschillende decision trees algoritme, gradient boosting algoritme, logistic regression algoritme en random forests algoritme die gebruikt worden voor diagnose van borstkanker. En er wordt voor een van deze modellen (algoritmes) gekozen.

# **Eisen**

* De nauwkeurigheid van de algoritme moet bijna 100% (of relatief dichtbij) zijn.

Dit is belangrijk om de aantal gevallen van *false negatives* en *false positive*  te verminderen.

* Data moet duidelijk en begrijpelijk gepresenteerd worden
* Hergebruik voor andere toepassingen die kunnen voordoen;
* Applicatie moet snel kunnen werken
* Algoritme moet goed gedocumenteerd

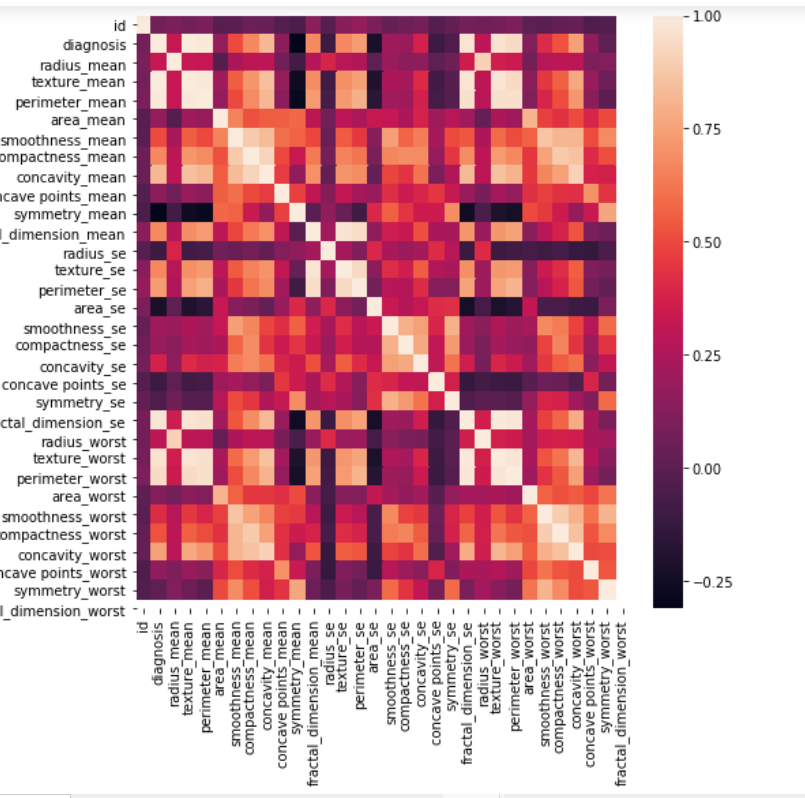
# **Algoritme**

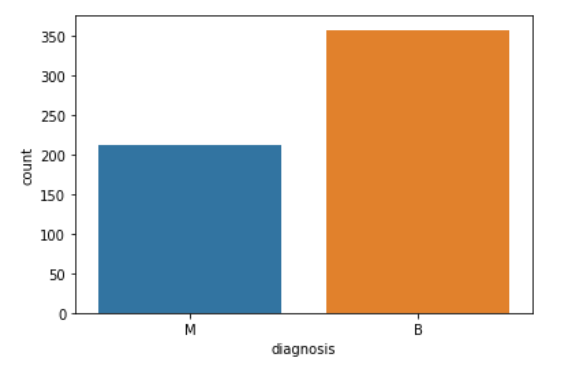
Aan de hand van een dataset met een diagnose van patiënten met borstkanker wordt een voorspellingsmodel voor het detecteren van borstkanker ontwikkeld. De dataset is afkomstig van dr. Wolberg en was opgenomen via de fijne naald aspiratie (FNA)[[1]](#footnote-0). Van de opgenomen monster van een cel via FNA, wordt 10 eigenschappen van de cel bepaald. Van elk van deze eigenschappen wordt *het gemiddelde*, de *standaardfout* en de '*worst' of slechte (gemiddelde van de drie grootste waarden)* berekend resulterend in een totaal van 30 eigenschappen. Deze eigenschappen vormen de basis van de dataset die gebruikt wordt voor het voorspellingsmodel.

In de eerste fase van het project wordt de dataset onderzocht. Door gebruik te maken van Pearson correlation wordt het verband tussen de variabelen in de dataset en de variabelen met de hoogste correlatie bepaald. Met het resultaat van de correlatie tussen de variabelen wordt aangegeven welke van de variabelen (eigenschappen van de dataset) een voorspellingswaarde heeft en welke geen voorspellingswaarde heeft.

Het resultaat wordt weergegeven via een Heatmap. De volgende variabelen hebben de hoogste correlatie met elkaar:

* perimeter\_mean en radius\_worst;
* area\_worst en radius\_worst;
* perimeter\_worst en radius\_worst, perimeter\_mean, area\_worst, area\_mean, radius\_mean;
* texture\_mean en texture\_worst;



Door gebruik te maken van de library Seaborn wordt deze variabelen met de hoogste correlatie gepresenteerd. 

Het resultaat van de diagnose in de dataset wordt visueel gepresenteerd. Figuur 1 geeft de presentatie hiervan. Uit deze dataset van 569 patiënten blijken 357 borstkanker goedaardig (benign) te zijn en 212 kwaadaardig (malign). Dat betekent dat 37.3% van de diagnose van de 569 patiënten kwaadaardig zijn.

Hierna wordt de dataset gesplitst in een training (75%) en testing (25%) dataset. De training dataset heeft een bekende resultaat en wordt gebruikt om de voorspelling uit te voeren. De testing dataset wordt gebruikt om de voorspelling op ons model te testen. Voordat we de splitsing van de dataset uitvoert, wordt de kolom diagnose gecodeerd (encode) in de waarden 0 en 1. Hiervoor wordt de functie Labelencoder gebruikt. De splitsing wordt uitgevoerd door gebruik te maken van de train\_test\_split methode in de SciKit-Learn library in Python.

***# Splitting the dataset into the Training set and Test setfrom***

***X = df.iloc[:, 2:32].values***

***Y = df.iloc[:, 1].values***

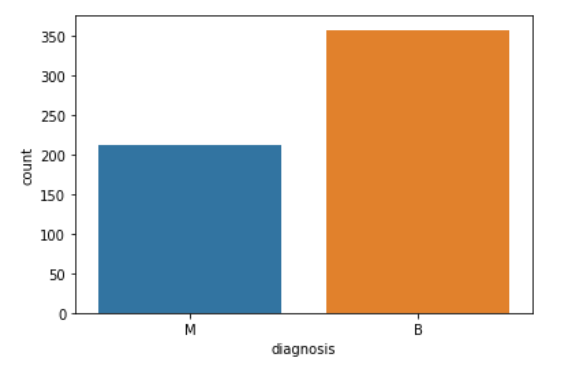
***labelencoder\_Y = LabelEncoder()***

***Y = labelencoder\_Y.fit\_transform(Y)***

***X\_train, X\_test, Y\_train, Y\_test = train\_test\_split(X, Y,***

***test\_size = 0.25, random\_state = 0)***

Om de dataset naar eenzelfde niveau van waarden te brengen, wordt gebruik gemaakt van schalen (normaliseren). De methode StandardScaler van SciKit-Learn library wordt hiervoor gebruikt.

Uiteindelijk wordt de algoritme gekozen om het beste resultaat te voorspellen. In de dataset hebben we de variabele or afhankelijke variabele Y met twee keuzes, namelijk M (Malign=kwaadaardig) of B (Benign=goedaardig). Daarom wordt gekozen voor Classificatie algoritme van supervised learning[[2]](#footnote-1). 

Er zijn verschillende methoden van classificatie algoritmes en ik heb gekozen voor Random Forest Classification.

De coding is als volgt:

***from sklearn.metrics import confusion\_matrix***

***cm = confusion\_matrix(Y\_test, model.predict(X\_test))***

***TN = cm[0][0]***

***TP = cm[1][1]***

***FN = cm[1][0]***

***FP = cm[0][1]***

***print(cm)***

***print('Model Random Forest Classifier Testing Accuracy = "{}!"'.format((TP + TN) / (TP + TN + FN + FP)))***

***print()# Print a new line***

***confusion = pd.DataFrame(cm, index = ['True', 'False'],***

***columns = ['negative', 'positive'])***

***#print confusion matrix in de vorm van een tabel***

***confusion***

Vervolgens wordt de diagnose van de borstkanker voorspeld door gebruik te maken van de test dataset.

***pred = model.predict(X\_test)***

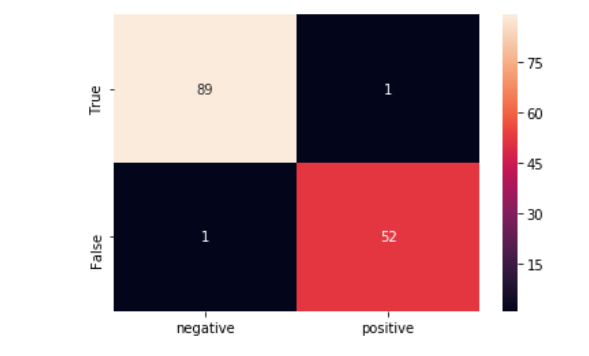
En vervolgens wordt de nauwkeurigheid berekent. Hiervoor wordt de confusion\_matrix gebruikt. De confusion matrix is een manier om de false classificaties te bepalen dus de geschatte diagnose die vallen in een verkeerde classificatie.

***sns.heatmap(confusion, annot= True)***

We gebruiken de classificatie nauwkeurigheid methode om de nauwkeurigheid van het model te bepalen. Met de nauwkeurigheid wordt bedoeld de verhouding tussen het aantal goed geschatte voorspellingen en de totale aantal input gevallen.

Het resultaat is: *Random Forest Classifier Training Accuracy: 99.76525821596243.* Dus 99.8% van nauwkeurigheid wordt bereikt.

De volgende grafiek presenteert ook het resultaat.



We kunnen concluderen dat ons voorspellingsmodel een goed model is om borstkanker te detecteren.

Ook wordt aan het model een naam gegeven. Het naam die gegeven wordt is TeTe. Het model is gemaakt op Aruba en daarom wordt een papiamentse naam ervoor gekozen, namelijk TeTe( = borst) .

1. Fijne Naald Aspiratie (FNA) is een diagnostische procedure die wordt gebruikt om klonters of massa's te onderzoeken. Bij deze techniek wordt een dunne (23-25 ​​gauge) holle naald in de massa gestoken voor het nemen van monsters van cellen die, na te zijn gekleurd, onder een microscoop worden onderzocht (biopsie). Biopsies met fijne naaldaspiratie zijn zeer veilige kleine chirurgische procedures. [↑](#footnote-ref-0)
2. Supervised learning is a type of system in which both input and desired output data are provided. Input and output data are labelled for classification to provide a learning basis for future data processing. Supervised learning problems can be further grouped into **Regression** and **Classification** problems. A **regression** problem is when the output variable is a real or continuous value, such as “salary” or “weight”. A **classification** problem is when the output variable is a category like filtering emails “spam” or “not spam”. Unsupervised Learning : Unsupervised learning is the algorithm using information that is neither classified nor labeled and allowing the algorithm to act on that information without guidance. [↑](#footnote-ref-1)